# Aligning to Human Decision-Makers in Military Medical Triage

Matthew Molineaux[1], Rosina O Weber[2], Michael W. Floyd[3], David Menager[1], Othalia Larue[1], Ursula Addison[1], Ray Kulhanek[1], Noah Reifsnyder[1], Christopher Rauch[2], Mallika Mainali[2], Anik Sen[2], Prateek Goel[2], Justin Karneeb[3] and JT Turner[3], and John Meyer[3]

[1] Parallax Advanced Research, Beavercreek OH 45431, USA,
first.last@parallaxresearch.org,
http://parallaxresearch.org
[2] Drexel University, PA, USA
[3] Knexus Research LLC, National Harbor, MD, USA

**Abstract.** Expert human decision makers do not make optimal decisions in realistic domains; their decisions are affected by preferences, ethics, background experience, and contextual factors. Often there is no optimal decision, or any consensus on what makes a decision good. In this paper we consider the problem of aligning decisions to human decision makers. To this end, we introduce a novel formulation of an aligned decision-making problem, and present the Trustworthy Algorithmic Delegate (TAD), an integrated AI system that learns to align its decision-making process to target decision-makers using case-based reasoning, Monte Carlo simulation, Bayesian diagnosis, and Naturalistic decision-making. We apply TAD in a military triage domain, where experts make different decisions, and present experimental results showing that it outperforms baselines and ablations at alignment in this domain. Our primary claims are that the combined components of TAD allows for aligned decision-making using a small, learned case base and that TAD outperforms simpler strategies for alignment in this domain.

**Keywords:** Case-based Decision-making, Alignment, Integrated AI, Case-based Agents

## 1 Introduction

In some decision-making domains, there is no consensus about how to judge correct or best decisions. Conflicting priorities, uncertainty caused by partial observations, time pressure, and diverse values lead to situations where experts often disagree on the correct action to be taken. When delegating decision-making to others in these domains, humans make reference to experience, ethics, and trust in choosing appropriate decision-makers. An important domain illustrating this kind of *difficult* decision-making is military medical triage, where human decision-makers must decide who will receive life-saving care when there is insufficient time and resources to save everyone [1]. For such contexts, there is no agreement on metrics for success, and no way to generate ground truth data evaluating decisions.

Because there is no object definition of success in these domains, we consider how to align to subjective definitions. In this work, we examine the problem of demonstrating alignment to a target profile made up of human decision-maker features. Such profiles assign values to a decision-maker along certain key features. For triage, examples are desire to maximize available information before making choices, and dedication to seeing that the "morally deserving" receive care first. We hope to allow non-experts the choice to delegate decisions to a system aligned to the decision-making model of trusted experts in emergency situations when an expert cannot be present.

The Trustworthy Algorithmic Delegate (TAD) system is a case-based agent designed to align to specific human experts on demand. The time of these experts is valuable, and the amount of data available for alignment is restricted. To do so, TAD uses case-based reasoning techniques that generalize well based on a small number of examples, leveraging robust similarity measures for retrieving relevant experience. Cases in TAD attach a decision-making *profile* that describes expert human decision-making characteristics to a situation and decision. At learning time, TAD creates a case base using available profile information for a group of experts and certain situations. Online, TAD estimates the profile of possible decisions by comparing to nearby cases in this case base.

A primary contribution of this work is the development of an autonomous CBR agent that performs user-aligned decision-making, which, to the best of our knowledge, has not previously been explored in the CBR literature.

In the rest of this paper, we: (1) detail the aligned decision-making problem and the triage problem, the context where TAD is described in this work; (2) discuss related work that solves similar problems; (3) describe the operation of TAD and its subsystems and components; and (4) describe experiments and results showing that the alignment of decisions produced by TAD in triage provides improvements over other decision-making strategies.

## 2 Aligned Decision Making Problem

In aligned decision making, there is no description of success in terms of goals or utilities (i.e., there is no single correct answer). Instead, there is a target decision-making *profile* to align to that represents characteristics of a human decision-maker, decision-making is judged based on how well a set of decisions satisfy that decision-making profile.

Formally, we characterize the general *alignable decision making problem* (ADMP) as a tuple $\langle S, S', A, \lambda, R(S'), P, pdet, pdist \rangle$. Given:

- a space of states $S$,
- a set of known starting states $S' \subset S$,
- a space of actions $A$,
- a transition function $\lambda : S \times A \to S$,

- a reachability function, $r : S \times S \rightarrow [0, 1]$ that separate states which have a trajectory between them in $\lambda$, and those that don't
- an alignment profile space $P$,
- a profile detection function over the reachable states $pdet : r(S') \times A \rightarrow P$, and
- a profile distance function $pdist : P \times P \rightarrow \mathbb{R}^{[0,1]}$,

Find: an *alignable decision maker* $\mathbb{ADM} : P \rightarrow (S \rightarrow A)$ that, given a profile $p \in P$, returns a policy $\pi : S \rightarrow A$ that minimizes

$$alignment(p, \pi) = \frac{\sum_{s \in S} pdist\left(pdet\left(s, \pi\left(s\right)\right), p\right)}{|S|}.$$

This function guides a learner toward reducing error between its responses and the responses dictated by a target profile. A key difference between the ADMP context and general optimal decision-making is that truth comes from the profile detection function *pdet* defined over actions rather than a goal or utility defined over states. We refer to decisions that minimize distance to a target profile as *aligned decisions*.

An important subclass of profiles represents humans considered to be experts by other humans; we refer to these as *admissible* profiles. An ADMP is *difficult* when there is a set of distinct admissible profiles $P_A \subseteq P$ such that for any pair of profiles $(p_1, p_2) \in P$, any policy $\pi_1$ that minimizes $alignment(p_1, \pi_1)$ cannot also minimize $p_2$, and vice versa. This can be thought of as two experts who have differing views on how a problem should be solved. If a single policy works for all experts, the ADMP devolves to an optimal decision-making problem.

$$\begin{aligned} difficult(S, S', A, \lambda, R(S'), P, pdet, pdist) \equiv \exists P_A \subset P, |P_A| > 1 : \forall p_1, p_2 \in P_A : \\ p_1 \neq p_2 \implies \forall \pi : \exists \pi' : alignment(p_1, \pi') < alignment(p_1, \pi) \\ \vee\, alignment(p_2, \pi') < alignment(p_2, \pi) \end{aligned} \quad (1)$$

### 2.1 Military Medical Triage

We investigate the difficult decision-making problem of military medical triage, specifically point of injury medical care. In the American military, personnel receive training to provide care beyond first aid, but have less medical education than a practicing doctor or medic. Decisions are life-changing, both for care providers and patients, and must be made rapidly, often in dangerous environments with limited resources. Personal accounts indicate that these are some of the hardest decisions to make and live with in a provider's entire life. We hope that in the future, tools for aligned decision-making will lift some part of these burdens.

In this work, we consider a simulated version of triage that allows exploration of this topic. Observations of the environment are complex, containing information about the world as well as the various people involved in the scenarios. In each state, there may be one or more patients, each with measurable vitals and possible injuries of various severity. The simulation supports 11 classes of injury and 14 types of medical supplies that can be used for treatments. Injuries can be immediately visible (e.g., amputation), discoverable (e.g., low blood pressure), or hidden (e.g., brain trauma). Seven characteristics make up patient vitals: consciousness, responsiveness to stimuli, ability to walk, mental status, breathing rate, heart rate, and blood oxygen level. In addition, the state includes a number of environmental factors, mission status, patient demographics, and patient status features. Actions include *situation report* (`SITREP`), which asks for patients' own description of their status, *vitals check* (`CHECK_ALL_VITALS`), in which all available vitals are recorded for a patient, *patient tagging* (`TAG_CHARACTER`), in which a triage tag is applied to a patient to indicate their status, and *treatment* (`APPLY_TREATMENT`), in which a medical procedure is performed to care for a patient (possibly using medical supplies). In `TAG_CHARACTER`, a triage tag describes injury severity and care prioritization, and includes *minimal* (i.e., minor injuries not requiring immediate care), *delayed* (i.e., serious injuries for which treatment can be delayed), *immediate* (i.e., severe, life threatening injuries requiring prioritized care), and *expectant* (i.e., unrecoverable injuries). In `APPLY_TREATMENT`, medical supplies to use for treatment must be selected, as well as the location on a patient's body to be treated (e.g., apply splint to left leg).

### 2.2 Specific Decision Maker Attributes

All decision-makers are assumed to have requisite skills to make good decisions (i.e., all trained in medicine), and each decision-maker attribute describes one aspect that affects decision selection when multiple reasonable decisions exist to choose from. In this work, we consider two attributes of triage decision-makers: *moral desert*, the degree to which a medic will prioritize patients whose moral culpability in injuries to others is lesser (i.e., the most morally deserving); and *maximization*, the degree to which a medic will prioritize actions that gain information over beginning treatment immediately [2].

An alignable decision-maker (ADM) does not attempt to achieve "correct" values on an attribute; instead, an ADM must be capable of aligning to a profile describing arbitrary features of human behavior. Decision-makers profiles contain values between 0 and 1 for each of a set of decision-maker attributes. Each profile may represent an individual decision-maker, cluster of decision-makers, or a theoretical decision-maker. Our current work has focused on decision-making profiles consisting of a single real-valued decision-maker attribute.

## 3 Related Work

The task of aligning an algorithmic decision maker with reference decision makers for contexts where a correct decision is unknown is relatively novel [3]. To address it, TAD innovates by incorporating other methods within CBR and by using data points obtained from those methods to add features to a case base.

CBR has a long history of integrations with other methods (*e.g.*, [4,5]). It is the way in which TAD integrates CBR with other methods that is unusual. Previously, due to machine learning underspecificity, [6] other work has proposed to obtain features using methods other than CBR for identifying new features to be added to a case base [7,8]. When doing this, [8] found in three different data types that average accuracy can be improved.

The topic of user preferences influencing decision making when multiple possible solutions exist has been previously explored using trust-guided behavior adaptation [9,10]. In this work, a CBR agent performs adaptive autonomy using both explicit and implicit feedback from a user. The agent uses an inverse trust estimate to keep a real-time measurement of its perceived trustworthiness and adapts its behavior when it believes its trust has fallen below a threshold. This work differs from ours in that it requires constant user feedback, which is unlikely to be available in high-pressure domains like medical triage.

The alignment profiles stored in our cases are similar to the idea of case provenance [11], where the source of a case is recorded such that more or less trust can be placed on it. The decision-maker attributes serve a similar role, allowing decision selection to be biased towards cases that come from decision-makers that more closely align with our desired behavior. However, in our work we do not explicitly record the case's source but instead use higher-level decision-maker attributes to record properties about the source.

## 4 Technical Approach

TAD's functions are divided across two distinct subsystems. The offline **training** subsystem (see Figure 1) learns case knowledge from observations and a profile detection function. The online **decision-making** subsystem (see Figure 2) uses the learned case base to select aligned decisions in response to observations. The online system is designed to operate quickly, and the offline system is designed to maximize performance. TAD components are orange, data input and output by TAD are blue; external sources (observations and the profile detection function) are shown in black.

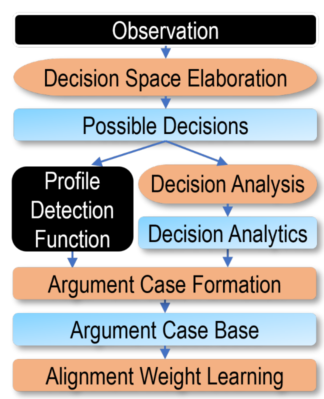


**Fig. 1.** TAD offline training subsystem creates the case base and weights

### 4.1 Subsystems

**Offline Training Subsystem** The TAD training subsystem uses examples of known user decisions to create cases. Based on the **Observations** of the environment (e.g., the state and any contextual information), **Decision Space Elaboration** generates a set of candidate decisions (**Possible Decisions**) that are feasible in the current situation. As an example, consider a situation with two patients with known injuries requiring treatment. Decision Space Elaboration would produce a set of possible `APPLY_TREATMENT` actions based on the patients, possibly injuries, and available medical treatments.

**Decision Analysis** considers features of a decision in various ways that are analogous to human considerations and produces **Decision Analytics** for each decision. *Monte Carlo Simulation* explores possible futures and effects of decisions; *Bayesian Diagnosis* considers hypotheses about what causes could underlie observations; and *Bounded Rationalizer* conducts fast and frugal methods to compare possible treatment and resource use decisions.

The **Profile Detection Function** is a black box that serves as a source of profile information during training. Profile information and decision analytics are fed into *Alignment Case Formation*, which outputs a case into the *Alignment Case Base*. Each case is made up of features describing a single observation, a decision made following that observation, decision analytics, and the decision-maker profile. After an entire case base is collected, the training subsystem performs **Alignment Weight Learning** to learn **Distance Function Weights** used to estimate decision-maker attributes.

**Online Decision-Making Subsystem** The decision-making subsystem reuses many similar TAD components to the training subsystem, with the primary difference being that the input Observations do not have known decisions associated with them (i.e., they are not training examples of past decision-making). Instead, the decision-making subsystem must select an appropriate aligned decision to perform based on a specified *Alignment Profile*. Each Observation is run through Decision Space Elaboration and Decision Analysis, similar to during training. The Observation, Possible Decisions, Decision Analytics, and Alignment Profile are used to query the Alignment Case Base (learned by the training subsystem) using the Distance Function Weights to find a case that is most similar to the current situation (i.e., Observation, Possible Decisions, and Decision Analytics) while aligning with the target decision-maker (i.e., Alignment Profile). The **Decision Selection** component then returns an **Aligned Decision** by selecting the decision that minimizes alignment distance, as estimated based on that case's neighbors.

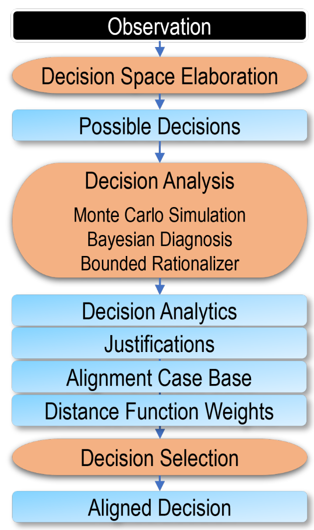


**Fig. 2.** TAD online decision-making subsystem makes aligned decisions

### 4.2 Components

**Decision Space Elaboration** TAD generates decisions in the triage domain by enumerating actions over all patients, treatments, categories, and injury locations. Each decision is checked to see whether it is compatible with the current state using domain-specific logic, and all reasonable decisions are retained. Nonsensical decisions (such as putting a tourniquet on a patient without a bleed) are removed. Future work will focus on extracting knowledge about reasonable decisions from domain documents.

**Decision Analysis** TAD analyzes decisions using analogues to how humans might consider them. It uses a variety of analysis techniques, each of which provides different types of analyses, which are run in parallel. We currently have implementations of three forms of decision analysis, described below: *Monte Carlo Simulation*, *Bayesian Diagnosis*, and *Bounded Rationalizer*. Each decision analysis component is primarily responsible for generating case features that describe aspects of a situation and possible decision analogous to what a human would consider in evaluating a possible decision. We enumerate these features for each component below; descriptions of these features may refer to a target patient or treatment specified by a decision. Because not all decisions involve treatments, features that refer to a treatment or target patient are left blank for decisions not involving them.
*Monte Carlo Simulation*:

A key factor in human decision making is considering the *effects* of our actions. Simplistically, that may only involve considering the immediate effects. However, there are also potentially delayed or longer-term effects. For example, consider having a small paper cut on your finger. One action that could be taken is to put an adhesive bandage over the cut, stopping the bleeding and protecting the wound. A potentially unforeseen impact of that action is that if there was only a single adhesive bandage, future treatments of additional injuries will not have adhesive bandages available.

To account for action effects, uncertainty, and longer-term impacts, we use a Monte Carlo Simulation approach. For each possible action that can be taken in the current environment state (i.e., from the Decision Space Elaboration component), a light-weight simulation is performed to measure the effects of taking that action. The Monte Carlo Simulation process performs the following steps:

1. Simulate performing a single action $a_0$ (i.e., the next action to perform) from among the set of possible actions to take.
2. Update the state from $s_0$ to $s_1$ based on performing action $a_0$.
3. Select another action $a_1$ to perform.
4. Update the state from $s_1$ to $s_2$ based on performing action $a_1$.
5. Continue selecting actions and updating the state until a completion criterion is reached (e.g., a fixed number of actions were performed).

Actions may have non-deterministic outcomes (e.g., applying a bandage may only stop bleeding some of the time, based on the severity of the cut) or the space of possible actions may be so large that considering all potential action sequences may be infeasible. Instead, Monte Carlo sampling is used where a set number of rollouts are performed, with each rollout examining a single action sequence given the non-deterministic action effects (i.e., a sequence $s_0$, $a_0$, $s_1$, $a_1$,

...). The results of all performed rollouts are combined to provide an estimate of an action's long-term effects given the environmental uncertainty that exists.

The Monte Carlo Simulation creates the following case features describing the expected state after a decision:

- `Target Patient Severity` (injury severity of the target patient)
- `Severest Severity` (injury severity of the most severely injured patient)
- `Overall Severity` (aggregate severity value for all patients)
- `Target Patient Severity Change` (expected change in Patient Severity)
- `Severest Severity Change` (change in Severest Severity)
- `Overall Severity Change` (change in Overall Severity)
- `Death Probability - Immediate` (probability at least one patient dies)
- `Death Probability - 1 min` (probability a patient dies within a minute)
- `Injury Per Second` ("amount" of injury occurring to patients each second: blood loss, lung damage, burn damage, and shock)
- `Supplies Used` (number of supplies used)
- `Supplies Remaining` (number of supplies remaining)
- `Weighted Supplies Used` (no. of supplies used weighted by importance)
- `Time Taken` (average time taken)
- `Information Gained` (reduction in uncertainty due to the action)

*Bayesian Diagnosis*: To handle uncertainty with respect to the current state of each patient, we use a Bayesian Network that describes the interrelation between various medical conditions and symptoms. This permits us to efficiently compute the joint posterior distribution of these variables given a set of (partial) observations provided by the triage simulator, which in turn allows us to provide two broad types of information to the Decision Selector:

- The probability of variables that are difficult or impossible to directly observe. This permits the operator to rapidly identify (and treat) immediately life-threatening conditions that are especially likely given the observations without first needing to perform additional (time-consuming) diagnostic actions.
- The entropy of either the network as a whole or of individual variables. By measuring the entropy before a proposed diagnostic action, and the *expected* entropy after, we can determine whether the expected reduction in uncertainty that the diagnostic provides is sufficient to justify the time it takes.

The conditional probability tables are currently rough estimates based on known causal links between medical conditions, and will be replaced by probabilities learned from records of trauma cases in the MIMIC-IV health record dataset on PhysioNet [12,13,14]. Inference is performed using the pyAgrum toolkit [15].

The Bayesian diagnosis analyzer outputs the following metrics for each action. Each is conditioned on the set of provided observations.

- `P(death)`: Probability of death if no further action is taken
- `P(pain)`: Probability of severe pain
- `P(brain injury)`: Probability of brain injury

- `P(airway)`: Probability of blocked airway
- `P(internal hemorrhage)`: Probability of internal hemorrhage
- `P(external hemorrhage)`: Probability of external hemorrhage
- `H(full)`: Entropy of the joint distribution over the full network
- `H(death)`: Entropy of the death random variable

*Bounded Rationalizer*: When faced with a vast amount of information, human decision-makers rely on heuristic decision-making and simplified search strategies rather than perfect rationality [16]. The Bounded Rationalizer simulates human-like automated search strategies that provide naturalistic solutions in general domains [17] and, in particular, medical decision-making [18].

Inputs for the different heuristics of the Bounded Rationalizer include decision feature selections, *predictors* of known preferences, and order of validity (for some heuristics). The Bounded Rationalizer generates all possible combinations of these inputs that could lead to a given decision. A set $M$ of *predictors* is used by the Bounded Rationalizer, features of a decision that may be central to its ranking with respect to other decisions. Predictors for patient priority include `age`, `rank`, `relationship`, and `injury_severity`. Predictors for treatment selection include `risk_reward_ratio`, `resources`, `time`, and `system`.

The Bounded Rationalizer uses case features to describe for each human search strategy and each selection of predictors, which patients and treatments are preferred. For each strategy, the case features are:

- `Target Patient Selection` is a single feature per strategy that describes whether the strategy would prioritize the target patient.
- `Treatment Selection` is a set of features per strategy that describes what treatments would be selected, given different predictor subsets $m \in M$.

Each strategy compares possible patients and treatments pairwise, then selects one or more options that maximize its heuristic. In the *exhaustive* strategy, the winner is selected based on a total ranking using all predictors in $M$. The *tallying* strategy considers predictors in $m$. In *take-the-best*, options are compared on predictors *in order of validity*, stopping as soon as one option has an advantage. The *satisfactory* strategy considers predictors in $m$ in a random order, stopping as soon as any predictor shows an advantage. In the *one-bounce* strategy, "potential winners" are found in random pairwise comparisons using all predictors in $M$. One-bounce selects the first potential winner that improves against $k$ other choices on the predictor subset $m$.

**Alignment Case Formation** Alignment cases are based on states, decisions, decision analytics, and attribute values. All decision analysis features described above are included in each case, as well as features of the state and decision. State features include:

- `unexamined_count`: Number of patients who have not been examined yet, other than the action target;
- `injured_count`: Number of injured patients, other than the action target;

– `others_tagged_or_uninjured_count`: Number of patients, other than the action target, who have received a triage tag or are uninjured;
– `aid_available`: True if evacuation is available or will be;
– `environment_type`: A high-level description of the treatment environment; one of *submarine*, *urban*, *desert*, or *jungle*.

Case action features are as follows:

– Action category:
  • `questioning`: Includes actions such as `SITREP` in which the medic talks to one or more patients.
  • `assessing`: Includes actions such as `CHECK_ALL_VITALS` in which the medic measures vital characteristics.
  • `treating`: Includes actions such as `APPLY_TREATMENT` in which the medic gives care to a patient.
  • `tagging`: Includes actions such as `TAG_CHARACTER` in which the medic applies a triage tag to a patient.
  • `leaving`: Includes actions such as `END_SCENARIO` which ends all further interaction with patients.
– Target patient demographics: `age`, `sex`, `rank`
– Target patient status:
  • `examined`: True if and only if a medic has checked the patient close enough to see visible injuries (typically after a `CHECK_ALL_VITALS`)
  • `tagged`: True if and only if target patient has a trauma tag
  • `relationship`: Describes an existing relationship between the medic and the patient; one of "friend", "relation", or "neutral"
  • `intent`: Describes whether the patient intended any participation in a harmful act. One of "intend major harm", "intend minor harm", "no intent", "intend minor help", "intend major help"
  • `directness_of_causality`: Describes a patient's role in causing an injury. One of "direct", "somewhat direct", "somewhat indirect", "indirect", or "none"
– patient vitals: `consciousness`, `responsiveness`, `ability_to_walk`, `mental_status`, `breathing_level`, `heart_rate`, `blood_oxygen_level`
– `category`: Triage tag category applied during a tagging action; one of "minimal", "delayed", "immediate", or "expectant"
– `treatment`: Specific treatment given during a treating action; includes 14 types medical supplies which can be applied to different areas of the body

**Alignment Weight Learning** CBR research and practice has long incorporated weighted similarity measures, distinguishing it from vanilla forms of k-nearest neighbors (kNN) algorithms. Seminal work from the 1990s [19,20,21] has established that feedback methods (*e.g.*, [22] those that assess the output produced by intermediary weights as feedback) are likely to lead to higher retrieval accuracy. Methods such as gradient descent [23], Relief [22] and its variations [24], decision trees [25], and genetic algorithms [26] have been used as alternatives for learning to represent the relative relevance of features for case-based similarity [27]. However, in the past 10 years, gradient boosted decision trees have substantially improved the performance of decision trees (*e.g.*, [28,29]). XGBoost [28] was found to improve accuracy of CBR retrieval when adopting as weights the importance factors from XGBoost in a regression task [30]. Our preliminary experiments confirm the higher accuracy of XGBoost over alternatives in TAD's classification task.

**Decision Selection** For each possible decision, the Decision Selector constructs a new case with features based on that decision, the current state, and the decision analytics. In current experiments, the decision-maker profile consists of a single decision maker attribute. To estimate the value of this attribute, the Decision Selector first finds the 4 nearest neighbors of each decision case as shown in Equations 2-4. The Euclidean distance function is computed with weights found by the alignment weight training component for the particular decision-maker feature.

$$nearestNeighbor(c, CB) = \underset{c' \in CB}{\arg\max}\, euclidean(c, c') \tag{2}$$

$$NN(c, CB, 0) = \emptyset \tag{3}$$

$$\begin{aligned} NN(c, CB, Size > 0) = {} & nearestNeighbor(c, CB) \\ & \cup NN(c, CB/nearestNeighbor(c, CB), Size - 1) \end{aligned} \tag{4}$$

The Decision Selector estimates the value of the target attribute by averaging the known attribute values of the nearest neighbors. Each neighbor provides a weighted contribution to this average based on its distance from the current case, as seen in Equations 5 and 6.

$$contribution(n, c, N) = \left( \sum_{n' \in N} \frac{euclidean(n, c)}{euclidean(n', c)} \right)^{-1} \tag{5}$$

$$\begin{aligned} & attributeValueEstimate(feature, c, CB) = \\ & \qquad \sum_{n \in NN(c, CB, 4)} contribution(n, c, NN(c, CB, 4)) \times feature(n) \end{aligned} \tag{6}$$

The Decision Selector chooses a single decision with a minimum difference between the estimated attribute value and the target attribute value.

## 5 Methodology and Results

We evaluated TAD in two settings, a *maximization*[1] and a *moral deserts*[2] setting. In each setting, we were provided with a profile detection function defined over a set of initial states in the space defined by our triage simulator. These states were partitioned into a training set and a test set in advance. We prepared our algorithms without knowledge of the test set. To train the case base, we recorded (1) the behavior of a random agent, (2) the agent's observations, (3) decision analytics for its actions, and (4) the profile detected for its actions by the provided profile detection function, on each of the training states 1,000 times. For each unique pair of observed state and selected action, we created a single case containing the state, decision, and decision analytic features corresponding to that unique pair, as well as two values concerning the decision-maker attribute profile found. One value was the detected attribute score, which had an undefined value for some state-action pairs; the other value was an approximation of the value of an action based on the detected attribute score of later actions. To approximate the action value, we averaged over the observed later outcomes of the action, discounting the late received value of an unscored action using an exponential decay (see Equation 7).

$$valueApprox(s_t, a_t) = \begin{cases} pdet(s_t, a_t) \neq \bot : & pdet(s_t, a_t) \\ pdet(s_t, a_t) = \bot : & 0.99 \times valueApprox(s_{t+1}, a_{t+1}) \end{cases} \quad (7)$$

After obtaining this case base, we found weights that predicted detected attribute scores without unscored actions (where *pdet* had an undefined value). To evaluate, we ran the TAD decision-making subsystem using the case base and weights created during training. Decisions were selected using the *approximated action values* of neighbor cases, as relevant neighbors did not always receive a defined score. We ran TAD for each scored initial state in the test set exactly once. For comparison, we ran two ablated versions of the system. The *informed baseline* used an ablated case base without the decision analytic features, and a uniform weighting scheme. The *uninformed baseline* considered the same decisions as TAD, but selected decisions based only on likelihood of survival, as determined by the projected "Overall Severity" value given by the Monte Carlo simulation, which describes how bad a state is after an action is taken. The informed baseline represents a naive attempt to approximate decision-maker attributes with no decision analysis. The uninformed baseline represents a typical decision-making system that does not attempt to align at all.

The alignment chart shows a visual depiction of how TAD and its two baselines compare on initial state groupings. Four scenarios are given for each setting,

---

[1]Initial states and profile detection function for maximization were provided by Alyssa Tanaka and colleagues with Soar Technology, LLC. Contact nicholas.paul@soartech.com to obtain.

[2]Initial states and profile detection function for moral deserts were provided by RTX BBN Technologies. Code can be downloaded from https://gitlab.com/itm-ta1-adept-shared/adept_server.

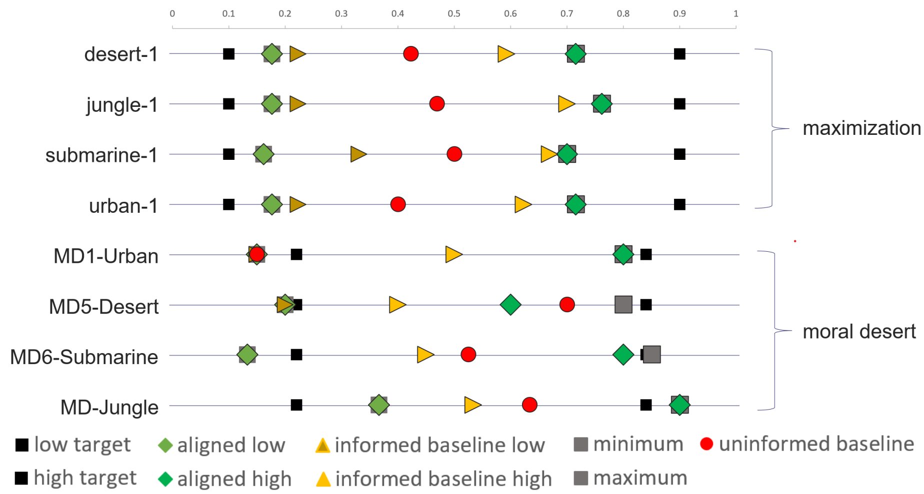


**Fig. 3.** Per-scenario alignment chart depicting the average maximization / moral deserts value attained by multiple versions of TAD

each with a set of initial states; 13 initial states are described in each group for maximization and 7 in each for moral desert. On each line, black squares give the target profile positions for each attribute on a 0 to 1 scale. Grey squares show the minimum and maximum values achievable on those initial states. Profiles detected for our three conditions are shown by different shapes; green diamonds for TAD, yellow triangles for the informed baseline, and red circles for the uninformed baseline. Both low (near 0) and high (near 1) targets are used in separate evaluation conditions, so we depict low and high values for both aligned TAD and the informed baseline. Since the initial states are the same for both evaluation conditions, and the uninformed baseline ignores the profile targets, only one value is shown for the uninformed baseline.

This chart gives an idea of how TAD and the baselines perform relative to one another at aligning to profile targets. In most cases, TAD's performance is closer to the profile target than either baseline, and often it is as close as possible (shown as a grey minimum or maximum square on top of green diamond in the chart). This gives an intuitive representation of what performance looks like.

To measure performance more objectively, we consider how often the decision picked by TAD in evaluation scenarios is maximally aligned to the target profile. The percentage of aligned decisions simply considers whether any alternative decision available to TAD had a profile closer to the target profile than the decision selected. We also consider the percentage achieved of possible alignment. Given a target profile $p^*$, this value is calculated as:

$$\sum_{s \in S, a \in A} \frac{pdist(p^*, pdet(s,a)) - \min_{a' \in A} pdist(p^*, pdet(s,a))}{\max_{a' \in A} pdist(p^*, pdet(s,a)) - \min_{a' \in A} pdist(p^*, pdet(s,a))} \tag{8}$$

Figure 4 shows the percentage of aligned decisions made by TAD and the two baselines. These are statistically significant differences, with TAD > Informed

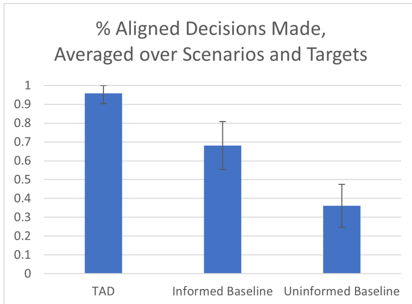


**Fig. 4.** % Aligned Decisions

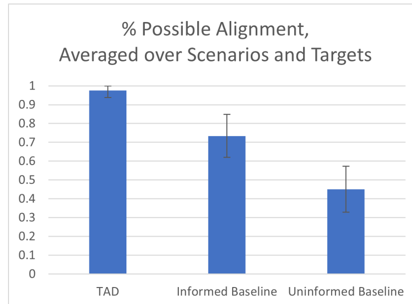


**Fig. 5.** % Possible Alignment

Baseline > Uninformed Baseline, with $p \approx .0037$. Figure 5 shows the extent to which each system matches the target profile. Results are different from aligned decisions because some misaligned decisions are closer to the target profile than others. For both figures, error bars indicate 95% confidence intervals over the mean for each condition.

## 6 Conclusions and Future Work

Experimental results so far are promising, showing the ability of a system with representations of human-like behavior to mimic opposed decision-making profiles given a relatively small amount of data. We can conclude that a naive approach is not sufficient to solve the problem. However, we have not shown generality, and future steps are necessary to make the TAD system approximate profiles in a larger range of domains.

We are enthusiastic about examining profiles based on multiple decision-maker attributes and their tradeoffs; data gathered so far does not support the study, but we expect to have such data soon. Local weighting strategies, where the presence of different context characteristics affect the distance function, are likely to come up with more complex attributes, and we expect to examine methods for acquiring local weights given sparse data. Ongoing work on case base augmentation will examine whether performance at aligned decision-making given sparse data can be improved by augmenting a case base with counterfactual data. Finally, we would like to reduce our dependence on knowledge engineering in decision space elaboration and the data analysis components, where human-generated probabilities, rules, and transition functions are used to help generate both possible decisions and decision analysis features.

*Code.* Code to replicate the experiments described in this paper can be downloaded from https://github.com/Parallax-Advanced-Research/ITM.

*Acknowledgements.* We'd like to thank scientists from RTX BBN Technologies and Soar Technology, LLC, who created the profile detection functions and datasets used in our experiments. We'd also like to thank CACI International, Inc., for creating the triage simulation used in our experiments.

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA) In the Moment program, contract number FA8650-23-C-7317. In the Moment (ITM) is assessing whether aligning AI to human values will make humans more willing to delegate decision-making in complex, time-sensitive situations where human experts disagree, such as battlefield triage. The views, opinions and/or findings expressed are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

## References

1. Shortland, N.D., Alison, L.J., Moran, J.M. *Conflict: How soldiers make impossible decisions*. Oxford University Press (2019)
2. Shortland, N., Alison, L., Thompson, L. Military maximizers: Examining the effect of individual differences in maximization on military decision-making. Personality and Individual Differences, vol. 163:p. 110051 (2020)
3. Turek, M. In the Moment (ITM) (2022). Contract Opportunity Type: Presolicitation (Original). BAA Number (Notice ID): HR001122S0031
4. Marling, C., Rissland, E., Aamodt, A. Integrations with case-based reasoning. The Knowledge Engineering Review, vol. 20(3):pp. 241–245 (2005)
5. Ali, R., Afzal, M., Hussain, M., Ali, M., Siddiqi, M.H., Lee, S., Kang, B.H. Multimodal hybrid reasoning methodology for personalized wellbeing services. Computers in biology and medicine, vol. 69:pp. 10–28 (2016)
6. D'Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B., Beutel, A., Chen, C., Deaton, J., Eisenstein, J., Hoffman, M.D., et al. Underspecification presents challenges for credibility in modern machine learning. Journal of Machine Learning Research, vol. 23(226):pp. 1–61 (2022)
7. Weber, R.O., Johs, A.J., Li, J., Huang, K. Investigating textual case-based XAI. In *Case-Based Reasoning Research and Development: 26th International Conference, ICCBR 2018, Stockholm, Sweden, July 9-12, 2018, Proceedings 26*. Springer (2018), pp. 431–447
8. Weber, R.O., Shrestha, M., Johs, A.J. Knowledge-based XAI through CBR: There is more to explanations than models can tell. In H. Borck, V. Eisenstadt, A. Sanchez-Ruiz, M. Floyd, editors, *Workshop Proceedings for the 29th International Conference on Case-Based Reasoning*. CEUR Workshop Proceedings (2021), vol. 3017, p. 103
9. Floyd, M.W., Drinkwater, M., Aha, D.W. How much do you trust me? Learning a case-based model of inverse trust. In *Proceedings of the 22nd International Conference on Case-Based Reasoning*. Springer (2014), pp. 125–139
10. Floyd, M.W., Aha, D.W. Incorporating transparency during trust-guided behavior adaptation. In *Proceedings of the 24th International Conference on Case-Based Reasoning*. Springer (2016), pp. 124–138
11. Leake, D.B., Whitehead, M. Case provenance: The value of remembering case sources. In *Proceedings of the 7th International Conference on Case-Based Reasoning*. Springer (2007), pp. 194–208
12. Johnson, A., Bulgarelli, L., Pollard, T., Horng, S., Celi, L.A., Mark, R. MIMIC-IV (version 2.2). PhysioNet. https://doi.org/10.13026/6mm1-ek67 (2023)

13. Johnson, A.E., Bulgarelli, L., Shen, L., Gayles, A., Shammout, A., Horng, S., Pollard, T.J., Hao, S., Moody, B., Gow, B., et al. MIMIC-IV, a freely accessible electronic health record dataset. Scientific data, vol. 10(1):p. 1 (2023)
14. Goldberger, A.L., Amaral, L.A., Glass, L., Hausdorff, J.M., Ivanov, P.C., Mark, R.G., Mietus, J.E., Moody, G.B., Peng, C.K., Stanley, H.E. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. circulation, vol. 101(23):pp. e215–e220 (2000)
15. Ducamp, G., Gonzales, C., Wuillemin, P.H. aGrUM/pyAgrum: a toolbox to build models and algorithms for probabilistic graphical models in python. In *International Conference on Probabilistic Graphical Models*. PMLR (2020)
16. Mosier, K.L. Searching for coherence in a correspondence world. Judgment and Decision Making, vol. 4(2):pp. 154–163 (2009)
17. Gigerenzer, G., Todd, P.M. Fast and frugal heuristics: The adaptive toolbox. In *Simple heuristics that make us smart*, Oxford University Press, pp. 3–34 (1999)
18. Marewski, J.N., Gigerenzer, G. Heuristic decision making in medicine. Dialogues in clinical neuroscience, vol. 14(1):pp. 77–89 (2012)
19. Wettschereck, D., Aha, D.W. Weighting features. In *First International Conference on Case-Based Reasoning*. Springer (1995), pp. 347–358
20. Wettschereck, D., Aha, D.W., Mohri, T. A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms. Artificial Intelligence Review, vol. 11:pp. 273–314 (1997)
21. Aha, D.W. Feature weighting for lazy learning algorithms. In H. Liu, H. Motoda, editors, *Feature Extraction, Construction and Selection: A Data Mining Perspective*, Springer US, Boston, MA, pp. 13–32 (1998). ISBN 978-1-4615-5725-8. doi: 10.1007/978-1-4615-5725-8_2
22. Kira, K., Rendell, L.A. A practical approach to feature selection. In *Machine learning proceedings 1992*, Elsevier, pp. 249–256 (1992)
23. Intelligence, B. Hepatitis diagnosis using case-based reasoning with gradient descent as feature weighting method. Journal of Information Systems Engineering and Business Intelligence: Volume 4 Number 1, April 2018, vol. 4:p. 25 (2018)
24. Kononenko, I. Estimating attributes: Analysis and extensions of relief. In *European conference on machine learning*. Springer (1994), pp. 171–182
25. Richardson, M.M., Warren, J.R. Induced decision trees for case-based reasoning. In *1996 Australian New Zealand Conference on Intelligent Information Systems. Proceedings. ANZIIS 96*. IEEE (1996), pp. 52–55
26. Chang, P.C., Lai, C.Y., Lai, K.R. A hybrid system by evolving case-based reasoning with genetic algorithm in wholesaler's returning book forecasting. Decision Support Systems, vol. 42(3):pp. 1715–1729 (2006)
27. Richter, M.M., Weber, R.O. *Case-Based Reasoning: A Textbook*. Springer, Berlin, Heidelberg (2013). ISBN 978-3-642-40166-4 978-3-642-40167-1. doi: 10.1007/978-3-642-40167-1
28. Chen, T., Guestrin, C. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd SCM SIGKDD international conference on knowledge discovery and data mining* (2016), pp. 785–794
29. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.Y. LightGBM: A highly efficient gradient boosting decision tree. Advances in neural information processing systems, vol. 30 (2017)
30. Ahmed, M.U., Barua, S., Begum, S., Islam, M.R., Weber, R.O. When a CBR in hand is better than twins in the bush. In *ICCBR XCBR'22: 4th Workshop on XCBR: Case-based Reasoning for the Explanation of Intelligent Systems at ICCBR-2022*. CEUR-WS.org (2022)